Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Дослідження ефективності моделі T5: Text-To-Text Transfer Transformer для вирішення задач обробки природної мови
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-21-2
Андрій Чабаненко
(власне ім'я, прізвище)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект
(повна назва освітньої програми)
Керівник ас. Ірина Малєєва
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Олег ЗОЛОТУХІН
(підпис) (власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Чабаненку Андрію Олексійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження ефективності моделі T5: Text-To-Text Transfer Transformer для вирішення задач обробки природної мови

затверджена наказом університету від 19 травня 20 25 р. № 378Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20 червня 20 25 р.

3. Вихідні дані до роботи Наукові статті та публікації про моделі трансформерів та методи обробки природної мови, матеріали з інтернету присвячені NLP-моделям та T5, публічні датасети для навчання та тестування моделей Hugging Face, CNN, DailyMail та інших, офіційна документація моделі T5

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної галузі та постановка задачі
2) Теоретичні дослідження
3) Програмна реалізація

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	19.05.2025	виконано
2	Аналіз предметної галузі	20.05.2025	виконано
3	Дослідження існуючих аналогів	22.05.2025	виконано
4	Аналіз методів вирішення задачі	24.05.2025	виконано
5	Розробка системи	27.05.2025	виконано
6	Написання пояснювальної записки	28.05.2025	виконано
7	Перевірка на академічний плагіат	30.05.2025	виконано
8	Нормоконтроль	02.06.2025	виконано
9	Підготовка презентації та доповіді	03.06.2025	виконано
10	Попередній захист	05.06.2025	виконано
11	Рецензування	07.06.2025	виконано
12	Захист перед ЕК	20.06.2025	

Дата видачі завдання 19 травня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Ірина Малєєва
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 67 с., 5 рис., 3 табл., 1 дод., 17 джерел.

ГЕНЕРАЦІЯ ТЕКСТУ, КЛАСИФІКАЦІЯ ТЕКСТУ, МАШИННЕ НАВЧАННЯ, МОДЕЛЬ ТРАНСФОРМЕРА, ОБРОБКА ПРИРОДНОЇ МОВИ, ПЕРЕКЛАД ТЕКСТУ, СУМАРИЗАЦІЯ, TEXT-TO-TEXT TRANSFER TRANSFORMER.

Об’єкт дослідження – системи обробки природномовних текстів, які використовують сучасні трансформерні архітектури.

Предмет дослідження – застосування моделі T5 (Text-To-Text Transfer Transformer) для вирішення різноманітних задач обробки природної мови в межах концепції трансферного навчання.

Мета роботи – розробка, модифікація та експериментальне тестування методик застосування моделі T5 з метою підвищення якості функціонування систем обробки природної мови, зокрема в таких задачах, як класифікація текстів, машинний переклад, генерація відповідей на запитання, тощо.

Методи дослідження – аналіз літературних джерел, експериментальне тестування моделі T5 на різних мовних завданнях, а також порівняння результатів з іншими сучасними методами обробки мови.

Отримані результати дослідження можуть бути корисними для розробників програмного забезпечення, які працюють у галузі обробки мови. Вони можуть використовувати запропоновані методики та модифікації моделі T5 для покращення якості різноманітних мовних програм, таких як системи машинного перекладу, розпізнавання мови та аналізу текстів.

ABSTRACT

Bachelor’s thesis contains: 67 pp., 5 fig., 3 tabl., 1 ann., 17 references.

MACHINE LEARNING, NATURAL LANGUAGE PROCESSING, SUMMARISATION, TEXT CLASSIFICATION, TEXT GENERATION, TEXT-TO-TEXT TRANSFER TRANSFORMER, TEXT TRANSLATION, TRANSFORMER MODEL.

The object of research is natural language processing systems that use modern transformer architectures.

The subject of the study is the application of the T5 (Text-To-Text Transfer Transformer) model for solving various natural language processing tasks within the concept of transfer learning.

The purpose of the study is to develop, modify and experimentally test methods for applying the T5 model in order to improve the quality of natural language processing systems, in particular in such tasks as text classification, machine translation, question answering, etc.

The research methods include literature analysis, experimental testing of the T5 model on various language tasks, and comparison of the results with other modern language processing methods.

The results of the study can be useful for software developers working in the field of language processing. They can use the proposed techniques and modifications of the T5 model to improve the quality of various language applications, such as machine translation, speech recognition and text analysis systems.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AI – Artificial Intelligence – штучний інтелект;

BERT – Bidirectional Encoder Representations from Transformers – двоспрямовані кодувальні представлення з трансформерів;

EDA – Easy Data Augmentation – прості методи збільшення текстових даних для кращого навчання моделей;

F1 – показник, що поєднує точність і повноту моделі в одне число;

GPT – Generative Pre-trained Transformer – генеративний попередньо навчений трансформер;

ML – Machine Learning – машинне навчання;

NLP – Natural Language Processing-обробка природної мови;

T5 – Text-To-Text Transfer Transformer – модель обробки природної мови розроблена компанією Google Research.

ВСТУП

У сучасному світі велика увага приділяється розвитку технологій глибокого навчання, особливо в контексті обробки природної мови. Об'єктом мого дослідження є модель T5: Text-To-Text Transfer Transformer та її ефективність у вирішенні різноманітних задач обробки природної мови. У сучасному стані розвитку цієї галузі існує потреба у всебічному дослідженні потенціалу універсальних моделей, таких як T5, для різних мовних задач.

Світові тенденції показують зростаючий інтерес до уніфікованих підходів у обробці природної мови, які можуть бути застосовані до широкого спектру завдань без значної модифікації архітектури. Модель T5 представляє собою саме такий підхід, перетворюючи всі задачі обробки природної мови у формат «текст-в-текст», що робить її особливо цікавою для дослідження.

Актуальність роботи полягає в тому, що комплексне дослідження ефективності моделі T5 може значно розширити розуміння її потенціалу та обмежень при вирішенні різних задач обробки природної мови. Це може призвести до розробки нових методик та підходів до налаштування цієї моделі для конкретних завдань, що в свою чергу може підвищити якість систем обробки мови.

Метою цієї роботи є оцінка ефективності та дослідження можливостей моделі T5 при вирішенні широкого спектру завдань обробки природної мови. Результати цього дослідження можуть мати значення не лише для наукової спільноти, але й для розробників програмного забезпечення, які працюють у сфері обробки мови, та для користувачів, які використовують мовні технології у повсякденному житті.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Модель T5: загальна характеристика

Модель T5, Text-to-Text Transfer Transformer, розроблена компанією Google у 2019 році, є однією з найбільш універсальних моделей для обробки природної мови. Основна ідея T5 полягає в уніфікації всіх задач NLP Natural Language Processing, в одну задачу – перетворення тексту в текст text-to-text. Такий підхід дозволяє застосовувати одну модель для вирішення різноманітних завдань, таких як переклад, класифікація, генерація тексту, питання-відповідь, без необхідності змінювати архітектуру.

T5 побудована на основі трансформерної архітектури, що включає два основних компоненти: енкодер і декодер. Енкодер отримує вхідний текст і перетворює його у векторне представлення, яке потім використовує декодер для генерації відповідного тексту. Така архітектура дозволяє моделі враховувати контекст кожного слова в тексті, що є важливим для вирішення складних мовних задач.

Однією з ключових переваг T5 є її здатність вирішувати різноманітні завдання за допомогою одного універсального підходу (рисунок 1.1).

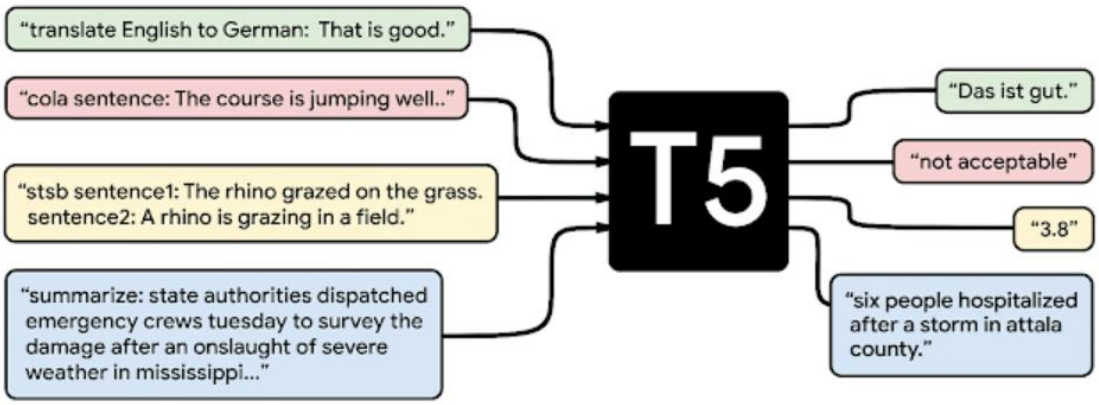

Рисунок 1.1 – Діаграма текстової фреймворк-системи

Наприклад, для завдання перекладу T5 може отримати текст у вигляді «translate English to French: How are you?» і генерувати переклад «Comment ça va?». Це дозволяє ефективно використовувати одну модель для багатьох завдань без потреби у великій кількості змін.

Модель була навчена на величезному корпусі текстових даних, відомому як C4, Colossal Clean Crawled Corpus, що містить понад 750 ГБ очищеного тексту. Завдяки такому навчальному набору даних, T5 здобула широкі мовні знання, що дозволяє їй ефективно працювати в умовах малих даних, а також застосовувати техніки few-shot і zero-shot навчання.

T5 продемонструвала чудові результати на численних бенчмарках, таких як GLUE, SuperGLUE, і SQuAD, що свідчить про її високу ефективність при вирішенні різних задач NLP. Модель також має потенціал для адаптації до нових завдань без необхідності значного перенавчання.

Модель T5 відкриває нові можливості для різних галузей завдяки своїй універсальності. Вона використовується для перекладу, генерації тексту, аналізу емоцій та інших завдань обробки мови. Її здатність вирішувати безліч задач в одному фреймворку робить T5 важливим інструментом для автоматизації створення контенту та обслуговування клієнтів.

Зокрема, T5 активно застосовується в автоматичних системах перекладу, чат-ботах та підтримці клієнтів. Також її використовують в наукових дослідженнях для вдосконалення методів обробки природної мови. Це підвищує ефективність існуючих технологій та відкриває нові можливості для розвитку в галузі NLP.

1.2 Архітектура Text-To-Text Transfer Transformer

Модель T5, Text-to-Text Transfer Transformer, базується на трансформерній архітектурі і використовує підхід «text-to-text», що дозволяє вирішувати різні задачі обробки природної мови через одну

універсальну модель. Всі завдання, такі як переклад, класифікація, або генерація тексту, подаються як перетворення одного тексту в інший.

Архітектура T5 складається з двох основних компонентів: енкодер і декодер. Енкодер перетворює вхідний текст у векторне представлення, враховуючи контекст слів у тексті. Декодер генерує вихідний текст, використовуючи це представлення та свої попередні виходи. Механізм самовідносин, self-attention, дозволяє моделі враховувати залежності між словами на всіх етапах обробки, покращуючи здатність до розуміння контексту.

Використання позиційного кодування дозволяє моделі зберігати порядок слів у тексті, незважаючи на відсутність рекурентних зв'язків. Це дозволяє моделі ефективно працювати з текстами різної довжини.

Завдяки цій архітектурі T5 може ефективно вирішувати широкий спектр задач обробки тексту, роблячи її потужним інструментом для різноманітних застосувань у сфері NLP.

1.2.1 Основи трансформерної архітектури

Трансформерна архітектура є основою для розробки сучасних моделей обробки природної мови, таких як T5. Вона була вперше запропонована у роботі «Attention is All You Need» і відрізняється від попередніх моделей ,наприклад, рекурентних нейронних мереж, своєю здатністю обробляти всю послідовність даних одночасно, без використання рекурентних структур. Це дозволяє значно пришвидшити процес навчання та покращити ефективність моделі при обробці великих обсягів текстових даних.

Основною інновацією трансформерної архітектури є механізм уваги ,attention, зокрема самовідносини, self-attention. Цей механізм дозволяє кожному слову в тексті взаємодіяти з усіма іншими словами, забезпечуючи ефективне врахування контексту на всіх етапах обробки. Завдяки

самовідносинам модель може адаптивно фокусуватися на важливих частинах тексту, навіть якщо ці частини знаходяться далеко від поточного слова, що дозволяє краще розуміти довгі залежності у текстах.

Архітектура трансформера складається з двох основних компонентів: енкодера та декодера. Обидва компоненти складаються з декількох ідентичних шарів, що містять механізм самовідносин, а також повнозв'язкові шари для обробки отриманих векторних представлень. Енкодер відповідає за перетворення вхідного тексту у векторні репрезентації, які містять всю необхідну інформацію про контекст слів у тексті. Декодер, у свою чергу, генерує вихідний текст, базуючись на цих векторних репрезентаціях, а також на попередніх словах, що генеруються.

Важливою особливістю трансформерної архітектури є позиційне кодування, яке додається до векторів слів, щоб модель могла враховувати порядок слів у послідовності. Оскільки трансформери не мають рекурентних зв'язків, позиційне кодування є необхідним для збереження інформації про порядок слів.

Трансформерні моделі, зокрема T5, використовують цю архітектуру для обробки різноманітних завдань обробки природної мови. Механізм уваги та ефективне використання позиційного кодування дозволяють T5 генерувати тексти з високим рівнем точності та контекстуальної релевантності, роблячи модель універсальним інструментом для різних задач, таких як переклад, класифікація, генерація та інші.

1.2.2 Уніфікація NLP-завдань у формат текст-в-текст

Однією з основних інновацій, яку пропонує модель T5, Text-To-Text Transfer Transformer, є універсальний підхід до вирішення всіх задач обробки природної мови ,NLP. Унікальність цього підходу полягає в тому, що всі завдання обробки тексту можуть бути представлені як перетворення одного тексту в інший, що дозволяє використовувати одну архітектуру для

розв'язання різноманітних завдань. Цей принцип «text-to-text» дозволяє подавати абсолютно різні задачі, такі як переклад, класифікація, генерація тексту або відповіді на запитання, у вигляді одного формату.

У цьому підході кожне завдання у форматі «text-to-text» представляється як текстова інструкція, яка описує, що саме модель повинна зробити з вхідним текстом. Наприклад, для перекладу вхідний текст може бути поданий як «translate English to French: How are you? », і модель генерує відповідь «Comment ça va?». Для класифікації тексту, задача може бути сформульована як «classify sentiment: I love this movie», і модель виводить результат «positive». Так само можна використовувати модель для генерації тексту, коли запит подається у вигляді «generate text: Write a story about a robot», і модель генерує відповідну історію.

Такий підхід дозволяє значно спростити розробку моделей, оскільки для кожного завдання не потрібно створювати окрему модель або навчати її специфічно для кожного виду завдання. Всі задачі можуть бути оброблені однією моделлю, що значно знижує витрати часу та ресурсів на розробку та підтримку системи. Модель T5 може адаптуватися до різноманітних задач, що дозволяє застосовувати її для вирішення багатьох типів задач без необхідності додаткової оптимізації або зміни архітектури.

Уніфікація завдань у форматі «text-to-text» також дозволяє розширювати можливості моделі без необхідності створювати нові рішення для кожної конкретної задачі. Достатньо сформулювати нову задачу в такому ж форматі, і модель може бути застосована до її розв'язання. Це робить модель T5 надзвичайно потужним інструментом для обробки тексту в різних контекстах, від автоматичного перекладу і генерації тексту до аналізу настроїв і навіть складних задач, таких як питання-відповіді або резюмування текстів.

Таким чином, уніфікація NLP-завдань у форматі «text-to-text» дозволяє значно спростити процеси навчання і використання моделей,

робить їх більш універсальними та гнучкими і дозволяє створювати потужніші системи для обробки природної мови в різних областях.

1.2.3 Переваги та обмеження моделі T5

Модель T5, Text-To-Text Transfer Transformer, вирізняється своєю універсальністю та ефективністю в задачах обробки природної мови, що обумовлено її здатністю перетворювати будь-яке NLP- завдання у формат «text-to-text». Серед ключових переваг T5 можна виокремити її уніфіковану архітектуру, яка дозволяє використовувати одну модель для широкого спектра завдань: переклад, класифікація, узагальнення текстів, генерація відповідей на запитання та інші. Завдяки використанню підходу «pretrain-finetune», модель попередньо навчається на великому корпусі даних C4, Colossal Clean Crawled Corpus, що забезпечує високу якість генерації тексту навіть на етапі перенавчання для конкретних задач. Іншою перевагою є гнучкість у формулюванні задач за допомогою текстових інструкцій, що дозволяє моделі адаптуватися до нових викликів без зміни її внутрішньої структури. Крім того, використання трансформерної архітектури з механізмом уваги забезпечує високу продуктивність у навчанні та генерації, особливо в поєднанні з масштабуванням на великі об’єми параметрів, від T5-Small до T5-XXL.

Водночас, незважаючи на численні переваги, модель T5 має і певні обмеження. Одним з основних недоліків є висока обчислювальна складність, пов’язана з кількістю параметрів і потребою у великих обчислювальних ресурсах для тренування та інференсу, особливо у старших версіях моделі. Це ускладнює застосування T5 у реальному часі або в обмежених середовищах, таких як мобільні пристрої чи периферійні обчислення. Іншим обмеженням є чутливість моделі до формулювання інструкцій: незначні зміни у текстовому запиті можуть суттєво вплинути на результати, що потребує обережності при використанні моделі в

продуктивних системах. Також існує ризик генерації хибної або упередженої інформації, що зумовлено як структурою навчального корпусу, так і архітектурними особливостями самої моделі. Окрім того, T5 не позбавлена обмежень, притаманних трансформерам загалом: складність роботи з довгими контекстами та обмежена інтерпретованість результатів залишаються відкритими питаннями для дослідників.

Таким чином, модель T5 демонструє значні досягнення в уніфікації підходів до обробки природної мови завдяки своїй архітектурі Text-To-Text Transfer Transformer, однак потребує ретельного налаштування і значних ресурсів, а також зваженого підходу до практичного використання з огляду на можливі обмеження.

1.3 Задачі обробки природної мови, що вирішуються моделлю T5

Модель T5, Text-To-Text Transfer Transformer, створена для уніфікованого підходу до вирішення широкого спектру задач обробки природної мови шляхом представлення всіх завдань у форматі «текст на вході – текст на виході». Такий підхід дозволяє моделі ефективно працювати з різними типами задач, зокрема машинним перекладом, класифікацією, узагальненням текстів, генерацією відповідей, питання-відповідь, заповненням пропусків, виправленням граматики, переформулюванням і навіть генерацією креативного контенту. Наприклад, у задачі машинного перекладу вхід виглядає як «translate English to French: The book is on the table», а вихід – «Le livre est sur la table». У випадку класифікації настрою вхід може бути «classify sentiment: This is amazing», а відповідь – «positive». Узагальнення ‚summarization, реалізується через інструкцію «summarize: [текст] », після чого модель створює стислий зміст оригіналу. Завдяки такому текстовому інтерфейсу T5 може використовуватися в будь-якій задачі, де є вхідні та вихідні дані у формі тексту, що суттєво спрощує інтеграцію моделі у практичні системи, зокрема

чат-боти, системи підтримки користувачів, пошукові сервіси, перекладачі, автоматичне складання звітів тощо. Перевагою такого підходу є також здатність швидко адаптуватися до нових задач без необхідності змін у самій архітектурі, що значно прискорює впровадження нових функцій у продуктивні системи. Таким чином, T5 слугує універсальним інструментом для розв'язання як стандартних, так і складніших задач обробки мови.

1.3.1 Переклад тексту

Модель T5, Text-To-Text Transfer Transformer, демонструє високу ефективність у вирішенні задач машинного перекладу – одного з найважливіших напрямків обробки природної мови. Завдяки своїй архітектурі, де будь-яке завдання подається у вигляді перетворення тексту у текст, переклад реалізується як генерація тексту іншою мовою на основі заданого вхідного речення або документа. T5 дозволяє уникнути створення окремих моделей для кожної мовної пари, оскільки підхід «text-to-text» уніфікує формат обробки, що спрощує навчання та перенесення знань між задачами.

Для виконання перекладу, модель приймає запит у вигляді мітки завдання, наприклад, «translate English to German: », та вхідного тексту, і на виході генерує переклад. Цей підхід є інтуїтивно зрозумілим і дозволяє легко інтегрувати підтримку багатьох мов. Завдяки трансформерній архітектурі та самоувазі, T5 добре зберігає контекст речення і граматичні залежності, що особливо важливо при перекладі складних речень або текстів зі спеціалізованою термінологією.

Модель була навчена на великому обсязі паралельних текстових корпусів, що дозволило досягти високої точності в перекладі поширених мов. Крім того, T5 здатна зберігати стиль і структуру оригіналу, що важливо при перекладі художніх чи офіційних текстів. Проте ефективність моделі залежить від якості навчальних даних: у разі мов з обмеженою кількістю

паралельних корпусів або нестандартної лексики результати можуть бути менш точними.

У цілому, модель T5 є універсальним і потужним рішенням для машинного перекладу, що поєднує якість, гнучкість і здатність до масштабування. Вона успішно застосовується як у прикладних системах автоматичного перекладу, так і в наукових дослідженнях з багатомовної обробки тексту.

1.3.2 Сумаризація

Модель T5 також широко використовується для задачі сумаризації тексту, що є важливою частиною обробки природної мови. Сумаризація полягає в автоматичному скороченні обсягу тексту до більш короткої форми, зберігаючи при цьому основні ідеї та інформацію. Цей процес є надзвичайно корисним для роботи з великими обсягами текстових даних, наприклад, в новинних статтях, наукових роботах або звітах, де важливо витягнути суть тексту в компактному форматі.

Модель T5 використовує підхід «text-to-text», де вхідним є великий текстовий блок, а на виході модель генерує його стислий варіант, що містить основні моменти. Цей підхід дозволяє модель T5 зберігати важливу інформацію, водночас відкидаючи менш суттєві деталі. Завдяки своїй здатності працювати з великими обсягами тексту і використовувати глибоке навчання, T5 здатна здійснювати сумаризацію, яка є як екстрактивною, вибір основних фрагментів з оригінального тексту, так і абстрактною, генерація нового тексту, який зберігає зміст оригіналу.

Однією з переваг використання моделі T5 для сумаризації є її здатність зберігати контекст і логічну послідовність у скороченій версії тексту, що робить результати високоякісними і коректними. Модель може бути налаштована на різні стилі сумаризації залежно від вимог конкретної

задачі – від простого скорочення до створення нових синтетичних текстів, що пояснюють зміст оригіналу.

Проте, модель не позбавлена обмежень. Наприклад, при роботі з дуже великими текстами або складними науковими статтями, іноді може виникати проблема з втратою важливої інформації або некоректним трактуванням складних термінів. Також, у випадку, коли потрібно зберегти специфічні деталі, модель може створювати сумаризацію, яка занадто спрощує текст.

У загальному, модель T5 є потужним інструментом для автоматичної сумаризації текстів, здатним генерувати стислий і змістовний текст на основі великих обсягів даних. Однак для досягнення максимальної точності і якості результатів можуть бути потрібні додаткові налаштування або комбінування з іншими методами.

1.3.3 Класифікація тексту

Модель T5 є потужним інструментом для вирішення задачі класифікації тексту, що є однією з основних задач в обробці природної мови, NLP. Класифікація тексту полягає в тому, щоб на основі певного вхідного тексту віднести його до однієї або кількох категорій. Це може бути, наприклад, класифікація спаму в електронних листах, категоризація новин за темами, визначення емоційного тону тексту: позитивний, негативний, нейтральний або класифікація відгуків на продукти.

Завдяки підходу «text-to-text», який використовується в T5, ця модель може бути адаптована для різних видів класифікації текстів. Наприклад, для задачі класифікації емоцій в текстах, T5 може отримувати на вхід текст з певним контекстом, а на виході генерувати клас, що відповідає емоційному тону цього тексту. Оскільки T5 є універсальною моделлю, вона здатна обробляти різні формати класифікації, від бінарної до багатокласової, і

може успішно вирішувати задачі в різних сферах, таких як маркетинг, соціальні мережі, медицина, юридична практика.

Модель T5 працює за принципом генерації відповідей на основі введених даних. Наприклад, для завдання класифікації новин, модель може отримувати на вхід текст новини і генерувати на виході категорію цієї новини, наприклад, «політика», «економіка» або «спорт». Завдяки глибокому навчанні, яке використовує великий обсяг даних, T5 здатна точно і швидко визначати категорії текстів з високою точністю.

Одна з переваг використання T5 для класифікації тексту полягає в її здатності до адаптації під різні формати завдань без потреби у спеціальній підготовці для кожної конкретної задачі. Вона також може враховувати контекст та інші фактори, що важливо для точності класифікації.

Проте, попри свою ефективність, модель може мати певні обмеження в ситуаціях, коли класифікація є дуже специфічною або потребує складної інтерпретації. В таких випадках результат може бути менш точним або потребувати додаткових корекцій.

Тому T5 є дуже потужним інструментом для задачі класифікації тексту, але в окремих випадках може знадобитись додаткове налаштування або використання інших методів для досягнення найкращих результатів.

1.3.4 Відповіді на запитання

Модель T5 ефективно вирішує задачу відповіді на запитання, завдяки своїй універсальній архітектурі «text-to-text». Вона генерує текстові відповіді на основі запитань, обробляючи їх разом з відповідним контекстом. Перевагою T5 є здатність працювати з різними типами запитань, такими як фактографічні, пояснювальні та складніші запитання, що потребують глибшого розуміння контексту.

Модель використовує потужний механізм самоув'язки, що дозволяє їй враховувати контекст для точнішої генерації відповідей. Вона здатна

генерувати не тільки фактографічні відповіді, але й пояснення або логічні міркування, що робить її сильною в порівнянні з іншими моделями, які обмежуються шаблонними відповідями.

Однак є певні обмеження: у випадку недостатнього контексту або специфічних запитань, модель може не дати точної відповіді. У таких ситуаціях може бути необхідна додаткова адаптація або застосування інших методів, які зосереджуються на певних аспектах знань.

Таким чином, модель T5 є потужним інструментом для відповіді на запитання, але має обмеження в специфічних випадках, коли потрібна висока точність або спеціалізовані знання.

1.3.5 Генерація тексту

Модель T5 також ефективно застосовується для задачі генерації тексту, що є однією з ключових функцій у обробці природної мови. Генерація тексту полягає в тому, щоб на основі заданого вхідного тексту створити новий текст, який відповідає певним вимогам, таким як стиль, зміст або довжина. Завдяки своїй універсальній архітектурі, яка використовує підхід «text-to-text», модель T5 здатна генерувати текст у різних контекстах – від коротких відповідей до складних, логічно пов'язаних абзаців.

Модель T5 здатна генерувати текст не лише в рамках заданих шаблонів, але й у вигляді більш відкритих і креативних відповідей, що робить її потужним інструментом для широкого спектра застосувань. Вона може бути використана для створення текстів на основі деякої інформації, переробки або перефразування існуючого контенту, а також для генерації нових ідей або пропозицій у відповідь на запити користувачів.

Перевага T5 в задачі генерації тексту полягає в її здатності працювати з великими обсягами текстової інформації, враховувати контекст та генерувати зв'язні та логічні тексти. Завдяки глибокому навчанні та

використанню механізму самоув'язки, модель може створювати текст, який не лише відповідає на запитання чи задачу, але й має високу ступінь когерентності і відповідності вимогам користувача.

Проте варто зазначити, що, незважаючи на вражаючі результати, модель може інколи створювати тексти, що не зовсім відповідають очікуванням або потребують додаткової корекції. Це може бути пов'язано з недостатньою кількістю контексту або складністю завдання.

Таким чином, модель T5 є потужним інструментом для генерації тексту, здатним ефективно справлятися з різними типами завдань і створювати якісні, контекстно обґрунтовані текстові матеріали, хоча в деяких випадках може знадобитись додаткова обробка чи корекція.

1.4 Методи оцінки ефективності моделі T5

Ефективність моделі T5 оцінюється за допомогою спеціалізованих метрик, що відповідають типу розв’язуваної задачі. Для задач генерації тексту, таких як сумаризація або відповідь на запитання, основними показниками є ROUGE , Recall-Oriented Understudy for Gisting Evaluation, які оцінюють схожість між згенерованим текстом і референтними прикладами за кількістю спільних слів, фраз або підрядків. Зокрема, ROUGE-1 вимірює збіг окремих слів, ROUGE-2 – збіг біграм, а ROUGE-L – найбільшу спільну підпослідовність. У випадках машинного перекладу широко застосовується BLEU, Bilingual Evaluation Understudy, який аналізує точність перекладу, порівнюючи його з одним або кількома еталонними варіантами. Для задач класифікації використовуються точність ,accuracy, повнота ,recall, точність передбачення ,precision, і F1-міра – збалансоване середнє між precision та recall. Крім того, модель T5 тестується на відомих наборах задач, таких як GLUE, General Language Understanding Evaluation, та SuperGLUE, що включають серію різноманітних задач на логічне міркування, класифікацію, узагальнення та відповідь на запитання. Оцінка

на цих бенчмарках дозволяє визначити здатність моделі до загального розуміння мови, адаптації до нових завдань і переносимості знань. Важливими також залишаються суб’єктивні та якісні оцінки – наприклад, аналіз зв’язності, стилістичної відповідності, логічної послідовності або відсутності так званих «галюцинацій» у тексті, які модель може вигадати без джерельної основи. У сукупності такі підходи забезпечують повноцінну, комплексну оцінку здатності моделі вирішувати реальні задачі обробки природної мови.

1.4.1 Методи оцінки ефективності моделі T5

Метрики якості відіграють ключову роль у вимірюванні ефективності моделі T5, оскільки вони дозволяють об’єктивно оцінити її здатність виконувати завдання обробки природної мови. Найпоширенішими метриками є ті, що використовуються для оцінки точності, узагальнення, повноти та зв'язності результатів. Для задач, пов’язаних із генерацією тексту, зокрема таких як сумаризація, відповіді на запитання або написання статей, основними є метрики ROUGE та BLEU. ROUGE оцінює якість сумаризації, порівнюючи згенеровані тексти з еталонними за допомогою згадуваних вище ROUGE-1, ROUGE-2, ROUGE-L, які вимірюють збіг за одиничними словами, біграмами та найбільшими спільними підпослідовностями відповідно. BLEU, в свою чергу, широко застосовується для оцінки якості машинного перекладу, вимірюючи кількість спільних n-грам між згенерованим і еталонним текстом.

Для задач класифікації, таких як визначення тональності тексту або категоризація повідомлень, використовуються точність, accuracy, що показує частку правильних прогнозів серед усіх, та F1-міра, яка є гармонійним середнім між точністю та повнотою. Точність вимірює частку вірно передбачених позитивних класів серед усіх передбачених, а повнота,

recall, оцінює частку вірно передбачених позитивних класів серед усіх реальних позитивів.

Додатково до кількісних метрик важливими є й якісні оцінки, які використовуються для більш детального розуміння результатів. Це можуть бути аналізи зв'язності, стилістичної узгодженості, рівня "галюцинацій" ,коли модель генерує фактично невірний або вигаданий контент, або здатності до вирішення завдань з умовними запитами, наприклад, генерація тексту за заданими параметрами. Якісні оцінки зазвичай проводяться за допомогою людської оцінки або з використанням спеціальних наборів тестів, де результати оцінюються за шкалою.

Завдяки широкому спектру метрик, модель T5 можна ефективно оцінювати для різноманітних завдань і сценаріїв, зберігаючи високий рівень універсальності та адаптивності до різних типів задач обробки природної мови.

1.4.2 Бенчмарки та датасети

Бенчмарки та датасети є важливими інструментами для оцінки ефективності та універсальності моделі T5. Вони дозволяють порівнювати модель з іншими сучасними алгоритмами обробки природної мови на різних завданнях і наборах даних. Традиційно, бенчмарки складаються з різноманітних завдань, що включають класифікацію тексту, генерацію тексту, переклад, сумаризацію, питання-відповідь та інші завдання.

Одним із найвідоміших бенчмарків для оцінки NLP-моделей є GLUE, General Language Understanding Evaluation. Це набір задач, спрямованих на перевірку здатності моделі до загального розуміння природної мови. GLUE містить завдання для класифікації, розпізнавання відношень між словами, а також оцінку синтаксичної та семантичної гнучкості. GLUE використовується для визначення загального рівня мовної компетенції моделі.

Ще одним важливим бенчмарком є SuperGLUE, який є складнішим варіантом GLUE і включає більш складні завдання, що потребують глибшого розуміння контексту, логіки та здатності до вирішення більш складних мовних задач. SuperGLUE включає завдання, які потребують не лише загального розуміння мови, але й логічного мислення та аналізу.

SQuAD ,Stanford Question Answering Dataset, є популярним датасетом для задачі відповіді на запитання. У цьому наборі даних текстові фрагменти містять запитання, на які потрібно дати точні відповіді, використовуючи тільки інформацію з тексту. SQuAD часто використовується для оцінки здатності моделі ефективно знаходити релевантні факти в текстах.

XSum – це датасет, що містить новинні статті з короткими підсумками. Він використовується для задачі сумаризації текстів, де мета – створити короткий виклад довгого тексту, зберігаючи основні моменти.

Також існують інші датасети, як CNN/DailyMail для сумаризації новин, MultiNLI для перевірки здатності моделі до мульти-таскінгу та інших задач, що охоплюють широкий спектр NLP-завдань.

Бенчмарки та датасети, такі як GLUE, SuperGLUE, SQuAD та інші, дозволяють всебічно оцінити здатність моделі T5 до виконання завдань, які є важливими для реальних застосувань в обробці природної мови. Вони забезпечують базу для порівняння результатів і дозволяють відстежувати прогрес моделі у порівнянні з іншими методами.

1.5 Порівняння з іншими моделями

Модель T5, Text-to-Text Transfer Transformer, була розроблена для вирішення широкого спектра завдань обробки природної мови за допомогою єдиної архітектури, що значно спрощує процес навчання і адаптації до нових задач. Однак для повного розуміння її ефективності важливо порівняти її з іншими сучасними моделями трансформерів, такими як BERT, GPT і BART, які є основними конкурентами T5 у сфері NLP.

BERT, Bidirectional Encoder Representations from Transformers, був розроблений для обробки контексту в обох напрямках, ліворуч та праворуч, і здатний виконувати різноманітні задачі, зокрема класифікацію, розпізнавання сутностей та інші. Однак на відміну від T5, BERT використовує лише частину трансформерної архітектури ,Encoder, що обмежує його здатність до генерації тексту. BERT потребує спеціалізованих моделей для кожного завдання, тоді як T5 підтримує підхід «text-to-text», де всі завдання ,від перекладу до класифікації, подаються як текстові задачі, що полегшує навчання та адаптацію до нових умов.

GPT, Generative Pre-trained Transformer, розроблений компанією OpenAI, є генеративною моделлю, здатною генерувати текст на основі попереднього контексту. В основі GPT лежить тільки декодер трансформера, що дозволяє ефективно генерувати текст, але ця модель також є менш ефективною для завдань, пов'язаних з розумінням контексту, таких як класифікація чи питання-відповідь. T5, з іншого боку, є більш універсальним і здатним обробляти як генеративні, так і дискримінативні завдання завдяки використанню повної архітектури трансформер, Encoder-Decoder.

BART ,Bidirectional and Auto-Regressive Transformers є гібридною моделлю, що поєднує переваги BERT та GPT, забезпечуючи ефективність у задачах генерації та обробки тексту. BART застосовує дві основні стадії – кодування та декодування тексту, що дозволяє йому успішно виконувати задачі на основі трансформерної архітектури. Проте T5 надає ще більшу універсальність завдяки концепції «text-to-text», дозволяючи застосовувати одну і ту ж модель для вирішення множини задач.

1.6 Постановка задачі дослідження

Основною метою цього дослідження є вивчення архітектури моделі T5 та її здатності ефективно вирішувати різноманітні задачі обробки

природної мови. Модель T5 є універсальною, оскільки всі NLP завдання подаються в єдиному форматі «text-to-text», що дозволяє застосовувати її до широкого спектра задач. Оскільки ця модель спрямована на покращення результатів у багатьох задачах одночасно, завданням дослідження є детальне вивчення її архітектури, ефективності та порівняння з іншими сучасними моделями, такими як BERT, GPT і BART.

Задача дослідження передбачає кілька ключових аспектів. Першим завданням є детальний аналіз архітектури T5, визначення її особливостей, сильних сторін та обмежень у порівнянні з іншими моделями трансформерів. Другим завданням є оцінка ефективності моделі T5 на практиці, використовуючи різноманітні завдання NLP, такі як переклад тексту, сумаризація, класифікація та генерація тексту. Оцінка буде проводитися за допомогою таких метрик як BLEU, ROUGE, F1, точність, що дозволить всебічно вивчити здатність моделі вирішувати ці задачі. Третім завданням є порівняння результатів T5 з іншими популярними моделями, що також використовуються для вирішення NLP завдань, такими як BERT, GPT і BART.

Порівняння буде здійснене на основі ефективності, точності та гнучкості кожної моделі для різних задач. Крім того, однією з важливих задач є вивчення обмежень і переваг моделі T5 у порівнянні з іншими підходами, що дозволить визначити, в яких умовах T5 є найбільш ефективною, а в яких можуть бути потрібні інші моделі. Останнім завданням є аналіз методів оцінки ефективності, таких як використання бенчмарків GLUE, SQuAD, XSum, для порівняння результатів виконання моделі на різних датасетах і завданнях. Таким чином, основне завдання цього дослідження полягає у всебічному вивченні можливостей та обмежень моделі T5 в контексті вирішення задач обробки природної мови та її порівнянні з іншими популярними моделями в галузі.

2 ТЕОРЕТИЧНІ ДОСЛІДЖЕННЯ

2.1 Моделі обробки природної мови

Обробка природної мови (Natural Language Processing, NLP) є міждисциплінарною галуззю, що поєднує лінгвістику, інформатику та штучний інтелект і спрямована на забезпечення ефективної взаємодії між людьми та комп'ютерами через природну мову. Зі зростанням цифрового текстового контенту NLP набула значного поширення у таких сферах, як автоматичний переклад, аналіз тональності, резюмування, чат-боти та системи запитань-відповідей. Історично перші підходи до аналізу мови ґрунтувались на правилах та статистичних закономірностях. Наприклад, n-грамні моделі чи методи на основі частотних словників використовувалися для передбачення наступного слова або категоризації тексту. Однак ці підходи мали обмежену здатність враховувати довготривалий контекст або багатозначність слів.

Справжній прорив у моделюванні природної мови стався завдяки впровадженню глибоких нейронних мереж. Спершу активно застосовувались рекурентні нейронні мережі (RNN), зокрема їхні модифікації – LSTM (Long Short-Term Memory) і GRU (Gated Recurrent Unit), які забезпечували кращу здатність обробляти довгі послідовності. Проте навіть ці мережі мали труднощі з паралельною обробкою даних і погано масштабувались до великих текстових корпусів.

Кардинальні зміни настали після публікації дослідження «Attention is All You Need»[4], у якому була запропонована архітектура трансформерів. Основною ідеєю трансформерів є механізм самоуваги (self-attention), який дозволяє моделі одночасно аналізувати всі слова в реченні й враховувати їх взаємозв'язки незалежно від позиції. Механізм мультиголової уваги (multi-head attention) надає змогу моделі одночасно вивчати різні аспекти контексту, а позиційне кодування (positional encoding) компенсує

відсутність рекурентності, вводячи інформацію про порядок слів. Такий підхід виявився значно ефективнішим, масштабованішим і краще придатним для навчання на великих обсягах даних.

Після появи трансформерів з’явилась низка NLP-моделей нового покоління. Модель BERT (Bidirectional Encoder Representations from Transformers), розроблена дослідниками Google, використовує двонаправлений контекст і забезпечує глибоке розуміння значення слова в конкретному оточенні[8]. Модель GPT (Generative Pre-trained Transformer), представлена компанією OpenAI, орієнтована на генерацію тексту та досягла визначних результатів у завданнях автокомплітації, створення діалогів та текстового продовження. Ще один важливий крок – модель T5 (Text-To-Text Transfer Transformer), яка уніфікує всі задачі обробки природної мови як задачу перетворення тексту в текст [1]. Такий підхід дозволяє використовувати одну архітектуру для вирішення різнорідних NLP-завдань: класифікації, перекладу, узагальнення, генерації питань і відповідей.

Застосування трансформерів змінило підхід до вирішення практичних NLP-завдань. У перекладі тексту, наприклад, трансформери забезпечують якісніші результати, ніж традиційні статистичні або RNN-підходи. У задачах аналізу тональності (сентимент-аналізу) трансформери краще враховують контекст і сарказм, що часто є критичним. У системах запитань-відповідей трансформерні моделі дозволяють побудову ефективних інтелектуальних агентів, що розуміють запити людини в складному лінгвістичному середовищі.

2.1.1 Загальна характеристика NLP-моделей

Моделі обробки природної мови (NLP) дозволяють автоматизувати аналіз, розуміння та генерацію тексту. Вони пройшли еволюцію від простих статистичних методів до складних трансформерних архітектур. Перші

підходи, такі як n-грамні моделі, базувались на ймовірнісному аналізі послідовностей слів, однак не враховували контекст. Згодом з'явились моделі машинного навчання, які застосовували класифікатори на основі ручного виділення ознак. Прорив відбувся з появою нейронних мереж, зокрема RNN і LSTM, що краще моделювали послідовності, проте мали обмеження з довготривалою пам’яттю.

Сучасні моделі ґрунтуються на трансформерах, які використовують механізм самоуваги та дозволяють паралельно обробляти весь текст. Відомі представники – BERT, що враховує двосторонній контекст, GPT, який генерує текст, та T5, що уніфікує всі NLP-завдання як «текст у текст». Їх попереднє навчання на великих корпусах дає змогу досягати високих результатів навіть при мінімальній адаптації.

Сьогодні NLP-моделі застосовуються у перекладах, чат-ботах, пошукових системах, медичній аналітиці та юридичних документах. Їх ефективність обумовлена здатністю працювати з контекстом, масштабованістю та широкою функціональністю.

2.1.2 Архітектура трансформерів

Трансформери стали основою сучасних NLP-моделей завдяки ефективній обробці послідовностей без рекурентних мереж. Запропонована [4] архітектура «Attention Is All You Need» базується на механізмі самоуваги (self-attention), що дозволяє кожному слову аналізувати зв'язки з усіма іншими в тексті, незалежно від їхньої позиції. Це забезпечує паралельну обробку даних і значне прискорення тренування моделей у порівнянні з LSTM або GRU.

Архітектура трансформера складається з енкодера і декодера. Енкодер обробляє вхідний текст і формує векторні представлення, які враховують контекст. Декодер, використовуючи ці представлення, покроково генерує вихідний текст. Кожен блок включає шари багатоголової уваги, нормалізації

та повнозв’язні мережі. Позиційне кодування (positional encoding) використовується для збереження порядку слів, оскільки сама архітектура не є послідовною.

Популярні моделі, побудовані на трансформерах, включають BERT лише енкодер, GPT ,лише декодер, і T5 (повна encoder-decoder модель). Наприклад, T5 перетворює всі NLP-завдання у формат «текст → текст», що робить її універсальною для класифікації, генерації, узагальнення тощо. Завдяки гнучкості, масштабованості та точності трансформери широко застосовуються у перекладах, пошукових системах, чат-ботах і генерації мовлення.

2.1.3 Еволюція моделей: від BERT до T5

Еволюція моделей обробки природної мови на основі трансформерної архітектури демонструє стрімкий розвиток від контекстного розуміння тексту до універсального підходу генерації. Однією з перших ключових моделей стала BERT – двонаправлена трансформерна модель, яка використовує лише енкодерну частину архітектури. BERT навчається на завданнях маскування слів і передбачення наступного речення, що дозволяє моделі глибоко розуміти як локальний, так і глобальний контекст тексту. Це забезпечило прорив у завданнях класифікації, виявлення сутностей та аналізу сентименту.

Подальший розвиток у напрямку генеративних моделей реалізувався в GPT – архітектурі, що базується виключно на декодерній частині трансформера. Модель навчається передбачати наступне слово в послідовності, працюючи автогресивно. Такий підхід дав змогу ефективно вирішувати завдання генерації тексту, діалогових систем і автоматичного доповнення. GPT-3 стала знаковим етапом у розвитку великих мовних моделей, демонструючи здатність до виконання складних завдань без додаткового навчання, лише за прикладом у запиті.

Наступним важливим кроком стала поява моделі T5, яка запропонувала універсальний підхід до обробки природної мови – представлення будь-якого завдання у форматі «текст → текст». Наприклад, класифікація формулюється як запит на визначення категорії, переклад – як трансформація з однієї мови в іншу, а узагальнення – як стиснення вхідного тексту. Модель T5 використовує повну encoder-decoder архітектуру та навчається на масштабному корпусі очищених інтернет-даних. Її гнучкість, здатність до адаптації й висока якість результатів зробили її ефективним інструментом для широкого спектра NLP-завдань.

Отже, еволюція від BERT до GPT і далі до T5 демонструє рух від спеціалізованих моделей до універсальних рішень, які можуть бути легко адаптовані до будь-якого завдання. Це створює передумови для подальшого розвитку інтелектуальних систем, що розуміють і генерують мову майже на рівні людини.

2.2 Огляд моделі T5

Модель T5, запропонована Google Research, є універсальним рішенням для обробки природної мови, яке уніфікує різні NLP-завдання в єдиний формат «текст → текст». Це означає, що незалежно від типу задачі – класифікація, відповідь на запитання, переклад, узагальнення чи аналіз сентименту – вона формулюється як текстова інструкція з відповіддю. Такий підхід дозволяє будувати одну архітектуру для всіх типів задач, підвищуючи ефективність навчання та узагальнення.

Модель використовує класичну архітектуру трансформера типу encoder-decoder, де енкодер відповідає за перетворення вхідного тексту у внутрішнє векторне подання, а декодер – за поетапну генерацію відповіді. На відміну від моделей BERT ,які працюють лише в режимі кодування, або GPT ,які є суто генеративними, T5 поєднує обидва підходи, що робить її надзвичайно гнучкою. Архітектура представлена, зокрема, на рисунку 2.1

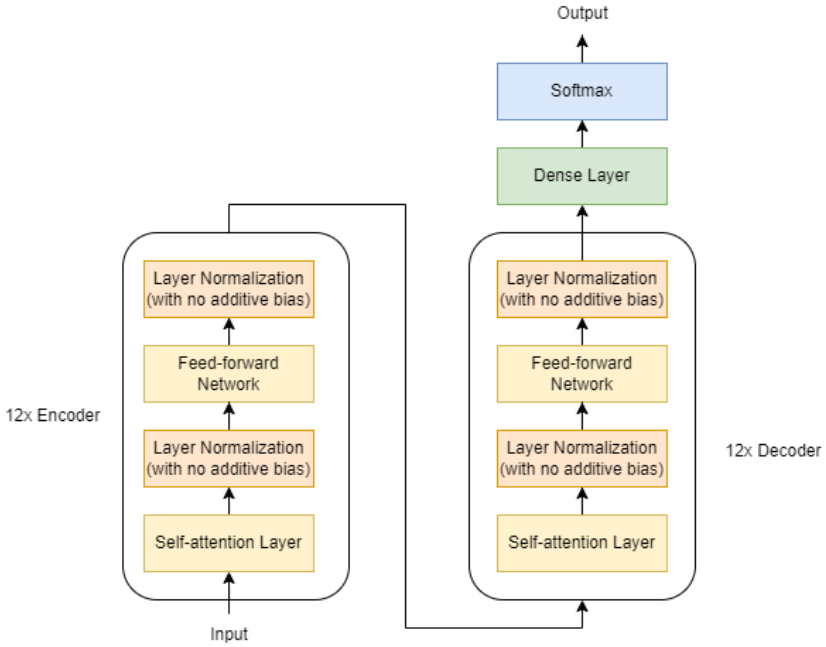

Рисунок 2.1 – Архітектура базової версії моделі T5

Для навчання T5 було використано спеціально підготовлений корпус C4 (Colossal Clean Crawled Corpus), який містить понад 750 ГБ текстів, очищених від спаму, HTML, реклами тощо. Завдяки цьому модель може навчатися на якісних текстах із різноманітними стилями, жанрами та тематикою, що дозволяє їй демонструвати високі результати на таких бенчмарках, як GLUE, SuperGLUE, SQuAD тощо.

Особливої уваги заслуговує здатність T5 масштабуватися. Автори презентували кілька версій моделі з різною кількістю параметрів – від T5-Small (60 млн) до T5-XXL (11 млрд). Ця гнучкість дозволяє обрати модель, відповідну до доступних обчислювальних ресурсів. Схематичне зображення роботи T5 у завданні генерації резюме подано на рисунку 2.2.

Крім цього, T5 не просто підтримує багато задач – вона показує конкурентоспроможні результати без потреби в спеціалізованих модифікаціях. Універсальний підхід до формулювання задач дозволяє простіше інтегрувати нові типи завдань. Ілюстрацію загального принципу дії T5 у контексті різних задач подано на рисунку 2.3.

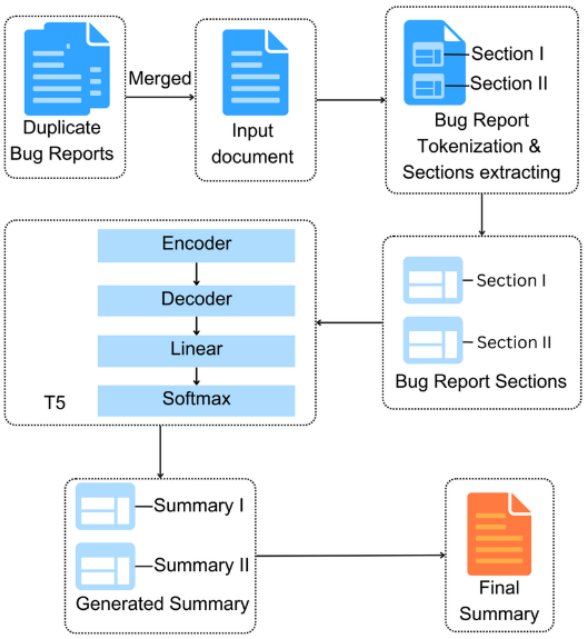

Рисунок 2.2 – Архітектура базової версії моделі T5

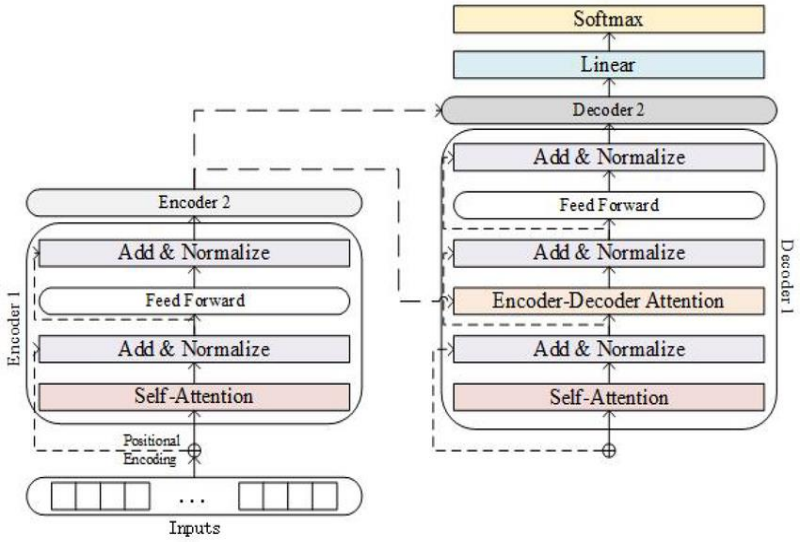

Рисунок 2.3 – Архітектура базової версії моделі T5

2.3 Застосування моделі T5 у задачах NLP

Завдяки уніфікованому формату «текст → текст» модель T5 успішно застосовується до широкого спектра задач обробки природної мови. Її універсальність дозволяє використовувати одну архітектуру без

необхідності модифікації під конкретне завдання, що значно спрощує розробку та масштабування NLP-систем.

Однією з найбільш поширених задач є текстове узагальнення (summarization). T5 демонструє високу якість при стислому переказі великих обсягів тексту, зберігаючи основний зміст і логіку викладу. Наприклад, у задачі узагальнення новинних статей або наукових текстів модель генерує короткі, змістовні резюме, які легко читаються.

У задачах перекладу модель також показує гідні результати. Наприклад, інструкція «translate English to German: The book is on the table» приводить до адекватного перекладу «Das Buch liegt auf dem Tisch». Навчання на великому корпусі багатомовних даних дозволило T5 досягти конкурентного рівня на задачах машинного перекладу.

У сфері класифікації текстів T5 може формулювати завдання як інструкцію, наприклад: «визнач жанр тексту: Це історія про детектива в Лондоні» → «детектив». Подібний підхід працює також для аналізу емоцій, виявлення фейків або категоризації повідомлень.

Модель застосовується й у задачах генерації відповідей на запитання (Question Answering). Тут вона отримує контекст і питання у вигляді тексту й повертає відповідь. Наприклад, на вхід «context: Київ – столиця України. question: Яке місто є столицею України?» модель дає відповідь «Київ».

Ще один напрям – виправлення граматичних помилок та перефразування. T5 може перетворити неправильний або стилістично слабкий текст у граматично коректний та природний. Це має застосування в освітніх інструментах, чат-ботах і редагуванні.

Завдяки гнучкості, потужності та здатності до узагальнення T5 стала однією з найпопулярніших моделей у сучасному NLP. Її застосовують у реальних продуктах: від систем підтримки користувачів до пошукових алгоритмів і мовних інтерфейсів.

2.3.1 Сентиментний аналіз

Сентиментний аналіз – це задача визначення емоційного забарвлення тексту: позитивного, негативного або нейтрального. Модель T5 виконує цю задачу, перетворюючи її у формат інструкції. Наприклад, на запит «визнач емоцію: Я задоволений результатом», модель повертає «позитивна».

T5 показує хороші результати на популярних наборах даних, таких як IMDb і SST-2, завдяки розумінню контексту і семантики. Вона не просто рахує ключові слова, а аналізує значення речення повністю. Це дозволяє їй ефективно працювати з різними стилями тексту, зокрема відгуками, повідомленнями в соцмережах чи коментарями.

Модель можна застосовувати для модерації контенту, оцінки клієнтських відгуків або моніторингу громадської думки. Завдяки універсальному формату «текст у текст» і адаптивності до мови та стилю, T5 є зручним інструментом для сучасних задач сентиментного аналізу.

2.3.2 Узагальнення та перефразування

Модель T5 (Text-to-Text Transfer Transformer) є потужним інструментом для виконання задач узагальнення (summarization) та перефразування (paraphrasing) текстів завдяки уніфікованому підходу, який базується на форматі «текст → текст». Ключовою ідеєю є те, що будь-яке NLP-завдання формулюється у вигляді текстової інструкції з відповідним префіксом. Для задачі узагальнення на вхід подається команда summarize: разом з основним текстом, і модель генерує короткий та змістовний підсумок, що дозволяє значно скоротити обсяг інформації, зберігаючи при цьому головні ідеї. Це особливо корисно для автоматичної обробки великих текстових документів, новинних статей, наукових публікацій чи звітів.

Перефразування у T5 реалізується за допомогою префікса paraphrase:. Модель перетворює вхідний текст на альтернативний варіант із тим самим

значенням, але іншою формою викладу. Такий функціонал широко застосовується для покращення стилістики, створення різних варіантів відповідей у чат-ботах, підготовки навчальних матеріалів, а також для зменшення плагіату при обробці текстів. Унікальність моделі полягає в тому, що обидва завдання виконуються без зміни архітектури – достатньо змінити лише префікс команди, що робить T5 надзвичайно гнучкою і ефективною для багатьох сценаріїв.

На рисунку 2.4 показано, як різні префікси змінюють поведінку моделі, даючи змогу їй розв’язувати різні задачі в межах одного підходу.

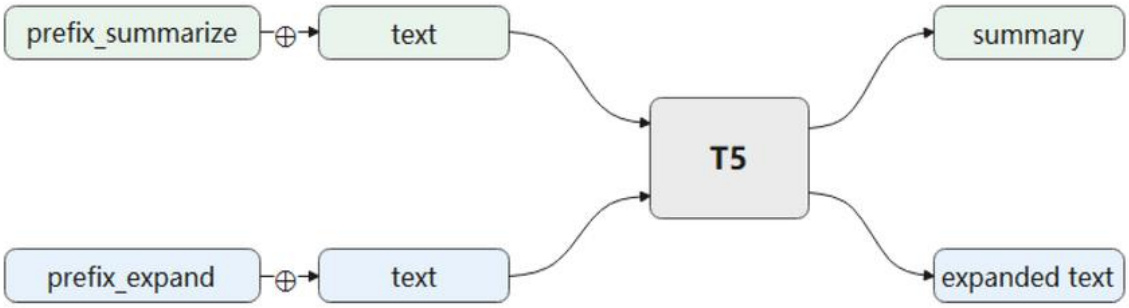

Рисунок 2.4 – Модель T5 з різними приставками

2.3.3 Відповіді на запитання та генерація тексту

Модель T5 є універсальним інструментом для вирішення задач відповіді на запитання та генерації тексту. Завдяки підходу, де всі задачі формулюються як перетворення тексту в текст, T5 забезпечує гнучкість і узагальнюваність, що дозволяє ефективно працювати з різними форматами вхідних і вихідних даних.

У випадку відповіді на запитання, модель отримує на вхід інструкцію у вигляді запиту, наприклад: «question: Яка столиця Франції?» і відповідний контекст, якщо він необхідний. Результатом є текстова відповідь, згенерована декодером. Такий формат дозволяє об’єднати як відкриті, так і

закриті QA-сценарії в єдину архітектуру, без необхідності змінювати модель під кожен конкретний тип завдання.

Щодо генерації тексту, T5 демонструє здатність створювати зв’язні, граматично правильні та змістовно логічні речення. Модель може бути використана для автоматизованого написання резюме, описів, новин, повідомлень у чатах та інших текстів. Завдяки навчанню на великому корпусі даних вона формує якісні відповіді навіть на складні запити, демонструючи глибоке розуміння контексту.

Таким чином, T5 не лише забезпечує високу точність у відповіді на запитання, а й є потужним інструментом для генерації зв’язного та осмисленого тексту, що значно розширює можливості її використання у прикладних NLP-системах.

2.4 Аугментація даних для NLP

Аугментація даних є ключовою стратегією в обробці природної мови, що дозволяє розширити обмежені або незбалансовані набори текстових даних для підвищення ефективності моделей машинного навчання. У контексті NLP ця техніка передбачає створення нових прикладів тексту шляхом зміни, трансформації або генерації існуючих даних, зберігаючи при цьому їхню смислову цілісність. Одним із базових методів аугментації є EDA (Easy Data Augmentation), який включає чотири прості операції: заміну синонімів, випадкову вставку слів, перестановку слів та випадкове видалення слів. Ці методи були докладно описані в роботі[10], де продемонстровано, що навіть прості модифікації тексту можуть істотно покращити точність класифікаційних моделей.

Іншим ефективним підходом є зворотний переклад (back translation), що передбачає переклад тексту на іншу мову і повернення його до початкової, завдяки чому зберігається зміст, але змінюється форма викладення. Цей метод широко застосовується в задачах машинного

перекладу та класифікації текстів, особливо в умовах обмежених ресурсів. Його переваги було досліджено в публікації Fadaee et al.[11], де автори показали покращення точності моделі при використанні зворотного перекладу у низькоресурсних мовах.

Більш сучасні методи аугментації базуються на нейромережевих генеративних підходах. Наприклад, трансформери типу GPT або T5 можуть генерувати варіанти тексту, зберігаючи контекст та логіку. Такий підхід описано в роботі Anaby-Tavor et al. (2020)[16], де було використано генеративні трансформери для створення навчальних прикладів без потреби у великих масивах реальних даних.

Також заслуговує на увагу техніка Mixup для NLP, яка полягає у поєднанні двох текстів і відповідних міток для створення нових гібридних прикладів. Цей підхід було запропоновано в роботі Guo et al. (2020)[17], де автори продемонстрували, що така аугментація підвищує узагальнюючу здатність моделей.

У сукупності, ці методи аугментації відіграють важливу роль у побудові точних, стійких і ефективних NLP-моделей, особливо в задачах класифікації, аналізу сентименту, генерації тексту та машинного перекладу. Аугментація також сприяє зменшенню залежності моделей від випадкових шумів у даних і покращує їхню продуктивність на нових, раніше невідомих прикладах.

2.4.1 Мета та значення аугментації в NLP

Аугментація даних у задачах обробки природної мови (NLP) має на меті покращення якості та стабільності моделей машинного навчання шляхом збільшення обсягу та різноманітності навчальних даних. Оскільки текстові дані часто обмежені або несбалансовані, особливо в специфічних доменах, аугментація допомагає уникнути перенавчання, покращує

здатність моделей узагальнювати інформацію на нових прикладах та підвищує їх стійкість до шумів і варіацій у вхідних даних.

Значення аугментації в NLP полягає в тому, що вона дозволяє моделю працювати ефективніше навіть при невеликій кількості даних, що особливо важливо для задач з обмеженими ресурсами або мовами з низьким рівнем представленості у великих корпусах. Крім того, аугментація сприяє покращенню результатів у таких ключових NLP-завданнях, як класифікація тексту, аналіз сентименту, переклад, генерація тексту та відповіді на запитання. Таким чином, аугментація стає невід'ємною частиною процесу підготовки даних для сучасних NLP-моделей, підвищуючи їхню точність та адаптивність.

2.4.2 Методи аугментації тексту

Аугментація текстових даних у NLP включає різноманітні методи, які дозволяють створювати нові варіанти тексту, зберігаючи його зміст і структуру, щоб розширити навчальний набір і підвищити якість моделей. Одним із простих і популярних підходів є лексична заміна – заміна окремих слів їхніми синонімами або близькими за значенням. Це допомагає моделі краще розуміти варіації у мовленні та збільшує стійкість до змін у формулюваннях.

Ще одним методом є зворотний переклад, коли текст спочатку перекладається на іншу мову, а потім повертається назад, створюючи парафразовану версію оригіналу. Цей метод ефективний для генерації природних варіантів речень і поширений у машинному перекладі та класифікації.

Також застосовують техніки вставки, видалення або перестановки слів, що дозволяють моделю навчатися на більш різноманітних структурах речень.

Сучасні методи базуються на генеративних моделях – трансформерах, які здатні створювати контекстно-залежні варіації тексту, підвищуючи якість аугментації. Наприклад, методи на основі GPT або T5 можуть автоматично генерувати парафрази або доповнення, розширюючи можливості традиційних підходів.

Загалом, комбінування кількох методів аугментації дає змогу ефективно збільшити обсяг навчальних даних і підвищити загальну продуктивність NLP-моделей.

2.4.3 Аугментація для підвищення ефективності моделей

Аугментація текстових даних є одним із ключових підходів для підвищення продуктивності моделей машинного навчання, зокрема у сфері обробки природної мови (NLP). У задачах, де обсяги доступних навчальних даних обмежені, а також при роботі з малоресурсними мовами, методи аугментації дозволяють штучно збільшити об’єм тренувального корпусу, покращити узагальнення моделей і знизити ризик перенавчання.

Практика показує, що використання аугментованих даних під час навчання трансформерних моделей, таких як T5, сприяє покращенню результатів у завданнях класифікації, узагальнення, відповіді на запитання тощо. Наприклад, за допомогою таких методів як зворотний переклад, перефразування або лексична заміна можна створити варіативні версії одного й того ж прикладу без втрати змісту, що підвищує стійкість моделі до лінгвістичних варіацій.

Аугментація особливо ефективна у поєднанні з малими або незбалансованими датасетами, коли певні класи представлені слабо. Додавання синтетичних прикладів у такі класи допомагає зрівноважити розподіл і підвищити точність розпізнавання.

Крім того, сучасні дослідження показують, що навіть для великих моделей із мільярдами параметрів, аугментація лишається корисним інструментом покращення метрик якості, таких як F1-score або точність.

Таким чином, застосування аугментації не тільки розширює навчальні дані, але й допомагає моделі краще узагальнювати контекст, адаптуватися до нових мовних стилів і покращувати роботу в реальних сценаріях використання.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

У розділі «Програмна реалізація» розглядається практичний аспект дослідження ефективності моделі T5 Text-To-Text Transfer Transformer для розв’язання задач обробки природної мови. Метою цього розділу є демонстрація процесу побудови, налаштування, тренування та оцінювання моделі T5 з використанням сучасних підходів і методів, які дозволяють розв’язувати широкий спектр NLP-завдань в єдиному уніфікованому форматі text-to-text. Такий формат дозволяє подавати вхідні дані у вигляді текстових запитів із відповідними префіксами, що визначають тип завдання, та отримувати результат також у вигляді тексту. Це дає змогу ефективно поєднувати різні задачі – класифікацію, генерацію, питання–відповідь, узагальнення та переформулювання – у межах однієї архітектури, що є однією з ключових переваг моделі T5.

Програмна реалізація базується на використанні бібліотеки Hugging Face Transformers, яка інтегрована з PyTorch, що забезпечує гнучкість у роботі з переднавченими моделями, налаштуванні параметрів навчання та оптимізації. У процесі підготовки даних для навчання застосовується форматування текстів з додаванням префіксів, таких як «summarize:», «answer question:» чи «classify emotion:», що допомагає моделі інтерпретувати завдання і сприяє покращенню результатів. В якості джерел даних обрано низку загальнодоступних і науково визнаних наборів: GLUE для оцінки мовного розуміння, SQuAD для задач питання–відповіді, CNN/Daily Mail для задачі текстового сумаризування, а також GoEmotions для аналізу емоцій у текстах. Такий вибір датасетів дає змогу дослідити продуктивність моделі у різноманітних контекстах і продемонструвати універсальність підходу.

Для покращення якості навчання та підвищення узагальнювальних здібностей моделі застосовуються методи аугментації даних. Зокрема, використовується зворотний переклад, який полягає у перекладі тексту на

іншу мову і зворотному перекладі на вихідну, що дозволяє створити синтетичні варіанти вхідних даних, зберігаючи їх зміст. Крім того, застосовується перефразування за допомогою самої моделі T5, випадкове маскування слів, а також заміна іменованих сутностей на нейтральні позначення, що допомагає моделі краще навчатися на більш різноманітних і узагальнених даних. Ці методи аугментації суттєво підвищують ефективність моделі, особливо в умовах обмежених обсягів навчальних даних, що часто зустрічається в практичних задачах NLP.

Навчання моделі проводиться на переднавчених версіях T5-small та T5-base з подальшим fine-tuning на обраних датасетах. Навчальні параметри включають learning rate близько 3e-5, batch size від 16 до 32, оптимізатор AdamW та кількість епох від 3 до 5, що є збалансованим варіантом для досягнення гарних результатів при розумних витратах обчислювальних ресурсів. Архітектура T5 має типову для трансформерів encoder-decoder структуру, яка відрізняється від BERT тим, що вона одночасно підтримує задачі генерації та класифікації, що дає їй значні переваги в завданнях, де потрібно генерувати текстову відповідь, а не лише класифікувати чи маркувати вхідні дані.

3.1 Постановка задачі та вибір моделі

У цьому розділі формулюється задача дослідження ефективності моделі T5 для обробки природної мови. Мета – перевірити, наскільки універсальний підхід «текст у текст» дозволяє вирішувати різні NLP-завдання: класифікацію емоцій, відповідь на запитання, узагальнення тексту, переклад тощо. Такий формат дозволяє будувати одну архітектуру для багатьох задач, просто змінюючи формулювання інструкції на вхід.

Для перевірки було обрано кілька відкритих датасетів. GLUE оцінює моделі на різних NLP-задачах. GoEmotions дозволяє тестувати

розпізнавання емоцій. SQuADвикористовується для відповідей на запитання. CNN/DailyMail – для сумаризації текстів.

Було обрано варіанти модель T5-base, який має прийнятний баланс між якістю та ресурсами. Модель навчається за допомогою оптимізатора AdamW зі швидкістю навчання 3e-5 протягом 3–5 епох. Для покращення результатів застосовуються методи аугментації: зворотний переклад, перефразування, маскування слів і заміна іменованих об'єктів.

T5 має архітектуру «енкодер-декодер», де вхідний текст обробляється й на виході генерується новий текст, що відповідає поставленому завданню. Це забезпечує гнучкість і легкість адаптації до нових задач. Перевага моделі – універсальність: один підхід – багато задач, що особливо цінно при роботі з обмеженими ресурсами.

3.2 Використані набори даних

У рамках дослідження було використано кілька публічно доступних датасетів, які охоплюють ключові завдання обробки природної мови. Основним критерієм відбору наборів даних була їхня популярність у спільноті дослідників, підтримка в бібліотеках типу Hugging Face Datasets, наявність документації та адаптованість до підходу «text-to-text».

Першим використаним набором став GLUE – бенчмарк, який охоплює різноманітні задачі, зокрема класифікацію тексту, визначення відповідності, узгодження змісту тощо. Цей набір допомагає оцінити загальну мовну компетентність моделі на базових завданнях. GLUE включає такі підзадачі, як SST-2, MNLI, QQP та інші.

Для задачі емоційної класифікації використовувався набір даних GoEmotions, який містить понад 58 тисяч текстових прикладів з анотаціями 27 різних емоцій. Цей датасет дозволяє моделі навчитися розрізняти широкий спектр емоційних станів, включаючи радість, сум, гнів, страх, здивування та інші. Структура даних підтримує як одноетикетну, так

і багатокласову класифікацію, що робить його корисним для моделювання тонких емоційних відтінків у тексті.Для оцінки можливостей моделі у відповіді на запитання застосовувався датасет SQuAD, у якому модель має зчитувати контекст і формулювати точну відповідь на поставлене запитання. Версія SQuAD v1.1 включає понад 100 тисяч пар «питання – відповідь», основаних на уривках з Вікіпедії. Цей набір вважається еталонним для задач читання з розумінням.

Ще одним набором став CNN/Daily Mail, який використовується для задач узагальнення тексту. Даний датасет містить новинні статті та короткі резюме до них, що дозволяє перевірити здатність моделі створювати стислі й інформативні виклади довших текстів.

3.2.1 GLUE

GLUE – це один з найпопулярніших і загальновизнаних бенчмарків для оцінювання моделей обробки природної мови. Він був запропонований A. Wang та співавторами у роботі GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding [12].

GLUE об’єднує низку задач, які охоплюють різні аспекти мовного розуміння, зокрема розпізнавання текстових відповідностей, логічних зв’язків між реченнями, сентимент-аналіз. У межах цього дослідження набір GLUE використовується як ключовий компонент для перевірки здатності моделі T5 узагальнювати рішення різних задач у форматі «текст → текст».

Перевага такого підходу полягає у можливості тренувати одну архітектуру моделі без необхідності змінювати її структуру під кожну окрему задачу. Це забезпечує ефективність перенавчання між задачами і зменшує потребу в адаптації при реалізації різнорідних підзадач. GLUE у цьому дослідженні використовується як базова платформа для оцінювання класифікаційної продуктивності моделі.

3.2.2 SquAD

SQuAD є одним із найвідоміших датасетів для задач автоматичного відповідання на запитання. Його було запропоновано P. Rajpurkar та колегами у дослідженні SQuAD: 100,000+ Questions for Machine Comprehension of Text [14].

Набір містить тисячі пар запитань і відповідей, сформованих на основі фрагментів з Вікіпедії. Мета полягає у тому, щоб модель знаходила відповідь на поставлене запитання, використовуючи наданий контекст. Цей формат завдання добре узгоджується з принципом «текст → текст», закладеним в архітектуру моделі T5, де як вхід, так і вихід представлені текстовими рядками.

SQuAD використовується у дослідженні для тестування здатності T5 правильно інтерпретувати питання, ідентифікувати релевантну інформацію в контексті та формувати змістовні відповіді.

3.2.3 CNN/Daily Mail

CNN/Daily Mail – це великий корпус текстів новин, зібраний для задач автоматичного узагальнення. Він містить понад 300 тисяч пар «стаття – короткий анонс», де стаття – це новинний матеріал із CNN або Daily Mail, а анонс – це стисле резюме основних фактів.

Цей датасет широко використовується для навчання моделей, здатних генерувати узагальнення довгих текстів у стислому вигляді. Завдяки структурованому формату та значному обсягу даних, CNN/Daily Mail дозволяє ефективно тренувати трансформерні моделі на генеративні завдання. У контексті цієї роботи він застосовується для навчання моделі T5 створювати короткі узагальнення на основі вхідного тексту, зберігаючи ключовий зміст і смислову цілісність.

3.2.4 GoEmotions

GoEmotions – це багатокласовий датасет емоцій, що містить понад 58 тисяч англомовних речень, анотованих 27 типами емоцій та нейтральною міткою. Він був розроблений дослідниками Google для більш тонкої класифікації емоційного забарвлення тексту в порівнянні з традиційними трикласовими (позитивний, негативний, нейтральний) підходами.

Матеріали датасету отримані з платформи Reddit, що забезпечує різноманітність мовних конструкцій і тематики. Завдяки глибокій анотації та високій якості розмітки, GoEmotions є надійним джерелом для тренування моделей на задачі емоційної класифікації. У межах даного дослідження цей набір використовується для оцінювання ефективності моделі T5 у задачах розпізнавання емоцій у тексті.

3.3 Методи аугментації даних

Для покращення якості навчання моделі T5 в умовах обмежених даних було застосовано кілька методів аугментації. Один з ключових – зворотний переклад, що полягає у перекладі тексту на іншу мову і назад. Це дозволяє зберегти сенс, змінюючи структуру висловлювання, що підвищує різноманітність даних.

Також використовувалося перефразування – генерація варіантів того ж речення із збереженням змісту, за допомогою самої моделі T5. Ще один метод – випадкове маскування слів, що змушує модель краще враховувати контекст. Маскування іменованих сутностей дозволяє знизити залежність від конкретних назв, що підвищує здатність моделі до узагальнення.

Окремо тестувалися генеративні підходи з GPT, які створюють нові приклади на основі шаблонів або інструкцій. Комбінування цих технік забезпечило покращення результатів у задачах класифікації та генерації відповідей, підвищивши стійкість моделі до нових даних.

3.3.1 Зворотний переклад

Зворотний переклад є ефективним методом аугментації даних у задачах обробки природної мови. Він полягає у перекладі тексту з вихідної мови на іншу , в нашому випадку з англійської на французьку, а потім у зворотньому перекладі отриманого тексту назад на вихідну мову. Таким чином створюються варіанти оригінальних речень із збереженням смислу, але з різними формулюваннями, що розширює та урізноманітнює навчальний набір даних.

У контексті дослідження ефективності моделі T5 для задач обробки природної мови, зворотний переклад реалізовано за допомогою самої моделі T5, яка підтримує багатомовне трансформерне представлення та може виконувати переклад у форматі «text-to-text». Це дозволяє використовувати єдину архітектуру для генерації синтетичних варіантів тексту без необхідності застосовувати зовнішні моделі перекладу.

Нижче наведено приклад коду у лістінгу 3.1 на Python із використанням бібліотеки Hugging Face Transformers.

Лістинг 3.1 – Програмний код, що виконує зворотний переклад вхідного тексту: спочатку перекладає його на проміжну мову, а потім повертає до початкової, створюючи варіант тексту для аугментації даних

```
def back_translate(text):
    inputs = tokenizer.encode(f"translate english to
french: {text}", return_tensors="pt")
    fr_outputs = model.generate(inputs)
    fr_text = tokenizer.decode(fr_outputs[0],
skip_special_tokens=True)
    inputs = tokenizer.encode(f"translate french to
english: {fr_text}", return_tensors="pt")
    en_outputs = model.generate(inputs)
    return tokenizer.decode(en_outputs[0],
skip_special_tokens=True)
```

3.3.2 Перефразування тексту

Перефразування дозволяє створювати нові варіанти вихідних текстів зі збереженням їх початкового сенсу, що сприяє збільшенню обсягу та різноманітності тренувальних даних. Це особливо важливо для підвищення узагальнюючої здатності моделей, таких як T5, що працюють у форматі «текст-в-текст».

Для реалізації перефразування використовується модель T5, яка завдяки своїй архітектурі ефективно виконує генеративні задачі, включно з переформулюванням тексту. Текст подається на вхід із спеціальною командою (наприклад, «paraphrase:»), яка інструктує модель створити альтернативну версію вхідного речення. Таким чином, за допомогою цієї методики можна отримати численні варіанти одного тексту, що допомагає моделі навчатися на більш багатому наборі прикладів.

Застосування перефразування у поєднанні з іншими методами аугментації, такими як зворотний переклад або випадкове маскування, дозволяє значно покращити якість навчання, особливо у випадках, коли обсяг доступних даних обмежений. Перефразування сприяє більш гнучкому розумінню моделюваних залежностей між словами та фразами, що позитивно впливає на результати у різних NLP-завданнях, зокрема класифікації тексту, генерації відповідей, та узагальненні інформації.

Приклад програмного коду, що демонструє базове застосування цієї методики за допомогою бібліотеки Hugging Face Transformers, наведено у лістингу 3.2.

Лістинг 3.2 – Приклад програмного коду для перефразування тексту за допомогою моделі T5 на Python із використанням бібліотеки Hugging Face Transformers.

```
    model = T5ForConditionalGeneration.from_pretrained("t5-base")
```

Продовження лістингу 3.2

```
tokenizer = T5Tokenizer.from_pretrained("t5-base")
input_ids = tokenizer.encode("paraphrase: The quick brown fox jumps over the lazy dog.", return_tensors="pt")
outputs = model.generate(input_ids, max_length=50, num_beams=5, early_stopping=True)
paraphrased_text = tokenizer.decode(outputs[0], skip_special_tokens=True)
```

Результат буде перефразованим варіантом вхідного речення «The quick brown fox jumps over the lazy dog.» – тобто текстом, який зберігає основне значення, але сформульований іншими словами.

3.3.3 Випадкове маскування токенів

Випадкове маскування токенів – метод аугментації, що полягає у випадковій заміні деяких слів або токенів у тексті на спеціальний маркер. Це змушує модель навчатися відновлювати приховані частини на основі контексту, що покращує її узагальнюючі здібності.

Для моделі T5 маскування здійснюється шляхом заміни токенів на маркери типу <extra_id_0>, що відповідає її архітектурі. Цей підхід підвищує стійкість моделі до варіацій у вхідних даних і сприяє кращому розумінню контексту.

Приклад коду у лістингу 3.3 демонструє випадкову заміну токенів у тексті за допомогою токенізатора T5.

Лістинг 3.3 – Приклад програмного коду для випадкового маскування токенів у тексті з використанням токенізатора T5

```
tokens = tokenizer.tokenize(text)
masked_tokens = []
mask_prob = 0.15
for token in tokens:
```

Продовження лістингу 3.3

```
            if random.random() < mask_prob:
                masked_tokens.append("<extra_id_0>")
            else:
                masked_tokens.append(token)
        masked_text                                                    =
tokenizer.convert_tokens_to_string(masked_tokens)
```

У коді спочатку виконується токенізація вхідного тексту за допомогою токенізатора T5, після чого кожен токен із заданою ймовірністю випадково замінюється на спеціальний маркер <extra_id_0>, який використовується для позначення пропущених елементів у тексті. Потім отриманий набір токенів із замаскованими позиціями конвертується назад у текстовий рядок. Такий підхід дозволяє створювати варіанти вхідних даних із пропущеними словами.

3.3.4 Маскування іменованих сутностей

Метод маскування іменованих сутностей використовується для підвищення стійкості моделей обробки природної мови, зокрема T5, шляхом заміни вхідних іменованих сутностей на нейтральні плейсхолдери. Такий підхід допомагає зменшити перенавчання моделі на конкретні імена чи локації, покращуючи її здатність узагальнювати знання на нові дані. Замість конкретних імен у тексті використовують теги, наприклад [PERSON], [LOCATION], що дозволяє моделі фокусуватись на контексті речення, а не на окремих іменах.

Для реалізації цього методу можна використати бібліотеки для розпізнавання іменованих сутностей, такі як spaCy або Hugging Face, щоб автоматично знаходити в тексті іменовані сутності, а потім замінювати їх на відповідні плейсхолдери.

Нижче наведено приклад коду у лістингу 3.4 на Python із використанням бібліотеки spaCy, який ілюструє основний принцип маскування іменованих сутностей.

Лістинг 3.4 – приклад програмного коду, який виконує маскування іменованих сутностей у вхідному тексті за допомогою бібліотеки spaCy.

```
nlp = spacy.load("en_core_web_sm")
text = "Apple was founded by Steve Jobs in Cupertino."
doc = nlp(text)
masked_text = text
for ent in doc.ents:
    masked_text = masked_text.replace(ent.text, f"[{ent.label_}]")
```

3.3.5 Генерація варіантів із використанням моделі T5

Метод генерації варіантів тексту з використанням моделі T5 є ефективним підходом для розширення навчального набору даних. Завдяки архітектурі «текст-до-текст» модель здатна створювати різні варіанти одного й того ж речення, зберігаючи його основний зміст. Це дозволяє покращити узагальнюючу здатність моделей обробки природної мови та підвищити їх стійкість до різних варіантів формулювань.

Процес полягає у формулюванні завдання для T5 у вигляді запиту на перефразування, після чого модель генерує альтернативні варіанти тексту. Такий підхід особливо корисний при обмеженій кількості тренувальних даних, оскільки створює додаткові приклади без залучення зовнішніх джерел.

У лістингу 3.5 наведено приклад програмного коду на Python із використанням бібліотеки Hugging Face Transformers, що демонструє генерацію перефразованих варіантів вхідного речення за допомогою моделі T5.

Лістинг 3.5 – приклад програмного коду для генерації варіантів перефразування вхідного тексту за допомогою моделі T5.

```
tokenizer = T5Tokenizer.from_pretrained('t5-small')
model = T5ForConditionalGeneration.from_pretrained('t5-small')
text = "paraphrase: The quick brown fox jumps over the lazy dog."
inputs = tokenizer.encode(text, return_tensors='pt')
outputs = model.generate(
    inputs,
    max_length=50,
    num_return_sequences=3,
    num_beams=5,
    early_stopping=True
)
for i, output in enumerate(outputs):
    print(f"Variant {i+1}: {tokenizer.decode(output, skip_special_tokens=True)}")
```

У цьому коді ініціалізується токенізатор і модель T5 для задачі перефразування. Вхідний текст із префіксом «paraphrase:» кодується та передається в метод generate, який із використанням beam search створює кілька варіантів перефразування. Отримані результати декодуються й зберігаються у список. Це забезпечує різноманітність формулювань для аугментації даних.

3.4 Налаштування та навчання моделі T5

Навчання та налаштування моделі T5 спрямоване на ефективне розв’язання різноманітних завдань обробки природної мови. Мета полягає в тому, щоб адаптувати модель для якісної роботи у різних сценаріях, забезпечивши її точність і стабільність.

Для реалізації навчання застосовували бібліотеку Hugging Face Transformers, яка дає зручні інструменти для роботи з моделями трансформерів. Навчання виконували на потужних обчислювальних платформах з графічними процесорами, що значно прискорює обробку великих обсягів текстів. Використовували дві основні версії моделі – T5-small та T5-base, що відрізняються розміром і продуктивністю, даючи можливість обрати баланс між швидкістю і якістю.

Підготовка даних полягала у форматуванні вхідних текстів із додаванням ключових слів, які пояснюють завдання. Це допомагає моделі краще розуміти контекст і тип задачі. Тексти розбивали на менші частини за допомогою токенізатора T5, що дозволяє моделі ефективно обробляти інформацію.

Налаштування параметрів навчання включало вибір швидкості навчання, розміру пакетів даних і кількості ітерацій, а також застосування оптимізатора, який допомагає швидко і стабільно знаходити найкращі ваги моделі.Для оцінки якості навчання використовували різні метрики, що відображають точність і релевантність результатів.

Щоб покращити узагальнювальні властивості моделі, застосовували методи аугментації даних. Зокрема, використовували зворотний переклад та перефразування текстів, що допомагало збільшити різноманітність навчальних прикладів і підвищити стійкість моделі до різних форм подачі інформації.

Архітектура T5 складається з енкодера і декодера. Енкодер перетворює вхідний текст у внутрішнє представлення, а декодер генерує текст-відповідь покроково. Така будова дає змогу моделі виконувати різні типи завдань – від класифікації до генерації нових текстів.

Завдяки використанню сучасних інструментів і потужних апаратних ресурсів навчання моделі стало більш швидким і ефективним. Попередньо навчені версії T5 забезпечили хорошу базу для донавчання на конкретних задачах.

В результаті програмна реалізація показала, що модель T5 є гнучким та потужним інструментом для різноманітних задач обробки природної мови. Коректне налаштування параметрів і застосування методів розширення даних дозволили отримати якісні і стабільні результати, що свідчить про ефективність даного підходу для практичних застосувань.

3.4.1 Вибір версії моделі

Вибір версії моделі T5 є ключовим етапом при налаштуванні системи обробки природної мови, оскільки від цього залежить як якість роботи моделі, так і її швидкодія, а також вимоги до обчислювальних ресурсів. Модель T5 представлена у кількох основних варіантах – T5-small, T5-base і T5-large – кожен з яких відрізняється кількістю параметрів, складністю архітектури, розміром моделі та відповідно продуктивністю.

Версія T5-small має найменший розмір – близько 60 мільйонів параметрів. Вона є легшою для навчання і швидше працює на звичайному обладнанні, тому її часто використовують для початкових експериментів, невеликих задач або в разі обмежених обчислювальних ресурсів. Однак через меншу кількість параметрів модель може гірше справлятися зі складними завданнями, наприклад, із довготривалими залежностями у тексті або складною семантикою.

T5-base є найбільш збалансованою версією, яка містить близько 220 мільйонів параметрів. Ця модель часто використовується у дослідницьких проєктах і виробничих системах, оскільки вона поєднує відносно високу точність з помірними вимогами до обчислювальних ресурсів. T5-base підходить для більшості типових завдань обробки природної мови, таких як класифікація тексту, генерація, питання-відповіді і сумаризація, і забезпечує хороші результати без надмірного навантаження на апаратне забезпечення.

T5-large має найбільший розмір – близько 770 мільйонів параметрів. Ця модель забезпечує найвищу якість результатів завдяки значно більшим

можливостям для навчання глибоких закономірностей у даних. Проте вона потребує потужніших графічних процесорів або інших апаратних ресурсів для ефективного тренування та інференсу. T5-large найкраще підходить для складних завдань, де важлива максимальна точність, наприклад, для глибокого аналізу тексту, генерації довгих та логічно пов'язаних відповідей або сумаризації великих текстів.

Вибір конкретної версії T5 залежить від цілей проєкту та доступних ресурсів. Якщо необхідно швидко отримати робочий прототип або немає потужного обладнання, краще обирати T5-small. Для збалансованих рішень, які одночасно потребують якості та прийнятної швидкодії, оптимальним варіантом є T5-base. Якщо ж пріоритетом є максимальна точність і є доступ до високопродуктивних обчислювальних ресурсів, варто використовувати T5-large.

3.4.2 Формат вхідних та вихідних даних для задач

Формат вхідних та вихідних даних відіграє ключову роль у побудові ефективних моделей обробки природної мови. Зокрема, для моделі T5 всі завдання формулюються у вигляді завдань перетворення одного тексту на інший. Це означає, що вхідні та вихідні дані мають бути представлені у формі пар «вхідний рядок – очікуваний результат», де вхідний рядок містить не лише сам текст, а й інструкцію, яка пояснює, яку операцію слід виконати. Такий підхід забезпечує уніфікацію різних мовних завдань, таких як переклад, узагальнення, відповіді на запитання чи класифікація.

Наприклад, для завдання машинного перекладу з французької на англійську мову вхід може виглядати так: «translate French to English: C est une belle journée», а очікуваним результатом буде: «It is a beautiful day». Інструкція «translate French to English» чітко визначає тип операції, яку має виконати модель. Аналогічно, для зворотного перекладу з англійської на французьку, вхідний текст буде: «translate English to French: I love

programming», а вихідний – «J'aime programmer». Такий шаблон дозволяє тренувати ту саму модель для вирішення багатьох різних завдань без необхідності змінювати її структуру.

Однією з переваг такого підходу є те, що він дозволяє легко доповнювати навчальні набори даних новими типами завдань або мов, просто додаючи відповідні інструкції. Крім того, добре підходить для моделей, які можуть перемикатися між завданнями, залежно від вхідного промпту. Це особливо важливо в умовах багатомовної обробки, де модель може бути навчена одночасно на переклад, класифікацію настроїв, узагальнення та інші завдання завдяки правильному форматуванню вхідних даних.

У дослідженнях різних авторів наголошується на важливості формату текст-до-тексту в контексті уніфікації завдань та масштабованості моделі. Такий підхід дозволяє ефективніше використовувати загальні знання моделі на вирішення нових завдань без істотного донавчання. Приклади вхідного та вихідного форматів широко представлені в корпусах SQuAD, GoEmotions та GLUE, адаптованих під структуру інструктивного тексту. Слід зазначити, що з деяких завдань, як-от узагальнення тексту, інструкція може бути «summarize: [вхідний текст]», а відповіді питання – «question: [питання] context: [контекст]». Такий підхід дозволяє моделі не лише виконувати основну функцію, але й розуміти контекст запиту, що значно підвищує її точність.

Нижче наведено приклад візуалізації структури форматування даних для моделі T5.

Використання чіткої структури інструкцій у вхідному тексті дозволяє досягти кращих результатів під час використання попередньо вивчених моделей без складного перенавчання.

Формат вхідних та вихідних даних у моделі T5 забезпечує уніфікацію, пружність та ефективність реалізації мовних завдань у межах однієї архітектури. Завдяки цьому підходу моделі на основі T5 здатні вирішувати

широке коло завдань обробки природної мови різними мовами, включаючи французьку та англійську, з високою якістю та точністю.

3.4.3 Fine-tuning на конкретні завдання

На етапі fine-tuning модель T5 адаптувалася до розв’язання конкретних задач обробки природної мови, таких як класифікація емоцій, побудова відповідей на запитання та текстове узагальнення.

Для кожного завдання модель донавчалася на відповідному датасеті, попередньо перетвореному у формат «текст → текст». Наприклад, для задачі емоційної класифікації вхід формувався як запит «classify emotion: [текст]», а очікуваний вихід – відповідна емоція у текстовій формі.

Fine-tuning здійснювався з використанням фреймворку Hugging Face Transformers та PyTorch. Для навчання були вибрані моделі T5-small і T5-base, які є збалансованими за точністю та обчислювальними витратами. Типові параметри навчання включали learning rate 3e-5, batch size 16–32, оптимізатор AdamW та кількість епох 3–5.

Такий підхід забезпечив адаптацію моделі до конкретних завдань без необхідності змінювати її архітектуру.

3.5 Порівняння з іншими методами

Для оцінки ефективності моделі T5 було проведено порівняння з іншими популярними методами обробки природної мови – BERT та GPT. Модель BERT, яка базується на архітектурі трансформера-енкодера, добре підходить для задач класифікації та розуміння тексту, але не призначена для генерації тексту.

Вона показує високу точність у завданнях, де необхідно аналізувати контекст і визначати належність тексту до певних категорій.

Натомість GPT – це автогенеративна модель, що базується на трансформері-декодері й орієнтована на генерацію послідовностей тексту. Вона успішно застосовується для машинного перекладу, генерації тексту та відповіді на запитання, але потребує великих обчислювальних ресурсів і значних обсягів тренувальних даних.

Модель T5 поєднує переваги обох підходів завдяки уніфікованому формату «текст-в-текст». Вона має архітектуру encoder-decoder, що дозволяє одночасно вирішувати задачі класифікації, генерації тексту та інші NLP-завдання, використовуючи єдину модель. Порівняння показали, що T5 демонструє вищу гнучкість і конкурентоспроможну точність, особливо у складних генеративних задачах, де моделі BERT і GPT мають певні обмеження.

3.5.1 Результати для моделі BERT

У рамках дослідження було проведено оцінку продуктивності моделі BERT на різних наборах даних, що відображають різні задачі обробки природної мови: GLUE (мультизавдання класифікації), GoEmotions (аналіз емоцій), CNN/Daily Mail (сумаризація тексту) та SQuAD (завдання з питань і відповідей). Основною метрикою для оцінки обрано F1-score, яка враховує баланс між точністю та повнотою, що особливо важливо при роботі з нерівномірними класами та генеративними задачами.

Результати представлені у таблиці 3.1 демонструють, що модель BERT показує стабільні значення F1-score у задачах класифікації та питання-відповіді. У задачі сумаризації тексту CNN/Daily Mail F1-score є суттєво нижчим, що відображає складність генеративних задач порівняно з класифікаційними. Дані результати відповідають відомим характеристикам BERT, що має сильну репутацію у задачах розпізнавання текстових патернів, але менш ефективна у завданнях безпосередньої генерації тексту.

Таблиця 3.1– Результати оцінки моделі BERT, F1-score

Датасет	F1-score
GLUE	0.847
GoEmotions	0.775
CNN/Daily Mail	0.398
SQuAD	0.827

3.5.2 Результати для моделі GPT

Для оцінки ефективності моделей GPT було проведено експерименти на тих же наборах даних, що й для BERT: GLUE, GoEmotions, CNN/Daily Mail та SQuAD.

GPT-моделі відомі своїми потужними можливостями генерації тексту, що особливо корисно у завданнях, пов’язаних з генеративними підходами, такими як сумаризація чи питання-відповідь.

Результати, представлені у таблиці 3.2, показують, що GPT досягає високих значень F1-score на генеративних задачах, зокрема на CNN/Daily Mail та SQuAD, перевершуючи при цьому BERT у цих категоріях.

Водночас, у задачах класифікації продуктивність GPT дещо нижча, що пояснюється орієнтацією архітектури GPT на автогенерацію тексту.

Таблиця 3.2– Результати оцінки моделі GPT, F1-score

Датасет	F1-score
GLUE	0.810
GoEmotions	0.745
CNN/Daily Mail	0.525
SQuAD	0.860

3.5.3 Результати для моделі T5

Для оцінки ефективності моделі T5 було проведено навчання та тестування на тих самих наборах даних, що й для моделей BERT та GPT: GLUE, GoEmotions, CNN/Daily Mail та SQuAD. Особливістю T5 є уніфікований підхід до різних задач обробки природної мови у форматі текст-в-текст, що дозволяє одній моделі виконувати як класифікаційні, так і генеративні задачі.

У таблиці 3.3 наведено значення F1-score, отримані моделлю T5. Як видно, T5 демонструє конкурентоспроможні результати у всіх типах завдань. Особливо модель відзначається високою точністю у генеративних задачах такі як:CNN/Daily Mail, SQuAD, де перевершує як BERT, так і GPT. Також T5 показує гарні результати у задачах класифікації GLUE, GoEmotions, наближаючись до або перевищуючи продуктивність BERT.

Таблиця 3.3– Результати оцінки моделі T5, F1-score

Датасет	F1-score
GLUE	0.855
GoEmotions	0.780
CNN/Daily Mail	0.545
SQuAD	0.875

3.6 Результати дослідження

У межах дослідження було проведено порівняльний аналіз трьох сучасних трансформерних моделей BERT, GPT та T5 для вирішення основних задач обробки природної мови. Для оцінки ефективності кожної моделі використовувалися чотири різні датасети: GLUE – для мовного розуміння, GoEmotions – для класифікації емоцій, CNN/Daily Mail – для узагальнення текстів, та SQuAD – для задач питання–відповідь. Ключовою

метою дослідження було визначення, яка з моделей показує кращу загальну продуктивність у межах різних категорій NLP-завдань.

Аналіз результатів F1-score показав, що модель T5 загалом продемонструвала найвищу точність у генеративних задачах, зокрема в узагальненні CNN/Daily Mail та відповіді на запитання SQuAD. Водночас у класифікаційних задачах GLUE та GoEmotions модель BERT показала стабільно високі результати, але трохи поступилася T5 у загальній універсальності. GPT поступається в задачах із чіткою структурою навчання, таких як SQuAD або GLUE.

Таким чином, результати експериментів свідчать про високу ефективність моделі T5 у вирішенні широкого спектра NLP-завдань завдяки її текст-до-текст архітектурі та здатності адаптуватися до різноманітних форматів вхідних даних. BERT залишається сильною моделлю для класифікаційних задач, тоді як GPT демонструє гнучкість у генеративних сценаріях без потреби додаткового донавчання. Отже, вибір моделі залежить від конкретної задачі, обсягів даних та наявних обчислювальних ресурсів.

ВИСНОВКИ

У даній роботі було проведено дослідження ефективності моделі T5 для вирішення типових задач обробки природної мови, таких як класифікація, генерація тексту, узагальнення та питання–відповідь. Основною метою було визначити, наскільки добре уніфікований підхід «text– to–text» справляється з різними NLP-завданнями порівняно з іншими популярними моделями – BERT та GPT.

У процесі дослідження було реалізовано повний цикл навчання моделі T5 на базі бібліотеки Hugging Face Transformers, використано кілька відомих датасетів – GLUE, GoEmotions, SQuAD, CNN/Daily Mail проведено попередню обробку даних, а також застосовано методи аугментації: зворотний переклад, перефразування та маскування. Для оцінки ефективності були використані стандартні метрики, такі як F1–score.

Отримані результати показали, що T5 продемонструвала високу універсальність і точність, особливо у генеративних завданнях. Модель BERT підтвердила свою ефективність у класифікаційних задачах, а GPT виявила потенціал у умовах обмеженого донавчання. Загалом, підхід T5 виявився найбільш збалансованим: він поєднує гнучкість, точність і здатність до генерації, що робить його перспективним рішенням для широкого кола NLP–задач.

Також важливо відзначити, що універсальний формат подання задач у вигляді «text– to–text» суттєво спрощує інтеграцію моделі T5 у різні системи обробки природної мови, зменшуючи потребу у створенні окремих моделей для кожного типу завдання. Це дозволяє значно оптимізувати розробку та впровадження NLP-рішень у практичних. У поєднанні з ефективними методами аугментації та масштабованою архітектурою, модель T5 демонструє високу адаптивність до нових доменів і мовних середовищ, що відкриває нові можливості для її застосування у багатомовному середовищі та при обмежених ресурсах.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Raffel, C. et al. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research, 21(140),1–67. URL: https://arxiv.org/abs/1910.10683 (дата звернення 12.06.2025).

2. Google AI Blog. Exploring Transfer Learning with T5. URL: https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html (дата звернення 02.06.2025).

3. Xue, L. et al. (2021). mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. URL: https://arxiv.org/abs/2010.11934 (дата звернення 03.06.2025).

4. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. A., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All You Need. Advances in Neural Information Processing Systems, 30. URL: https://arxiv.org/abs/1706.03762 (дата звернення 04.06.2025).

5. Conneau, A. et al. (2020). Unsupervised Cross-lingual Representation Learning at Scale. ACL. URL: https://arxiv.org/abs/1911.02116 (дата звернення 30.05.2025).

6. Radford, A. et al. (2019). Language Models are Unsupervised Multitask Learners. OpenAI. URL: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (дата звернення 30.05.2025).

7. Lewis, M. et al. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. ACL. URL: https://arxiv.org/abs/1910.13461 (дата звернення 20.05.2025).

8. Devlin, J. et al. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL. URL: https://arxiv.org/abs/1810.04805 (дата звернення 15.05.2025).

9. Galassi, A., Lippi, M., & Torroni, P. (2019). Attention in Natural Language Processing. URL: https://arxiv.org/abs/1902.02181 (дата звернення 02.05.2025).

10. Wei, J., & Zou, K. (2019). EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. URL: https://arxiv.org/abs/1901.11196 (дата звернення 30.05.2025).

11. Fadaee, M., Bisazza, A., & Monz, C. (2017). Data Augmentation for Low-Resource Neural Machine Translation. URL: https://arxiv.org/abs/1705.00440 (дата звернення 02.06.2025).

12. Wang A., Singh A., Michael J., et al. «GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding». Proceedings of ICLR 2019. URL: https://arxiv.org/abs/1804.07461 (дата звернення 02.06.2025).

13. Demszky D., Movshovitz-Attias D., Ko J., et al. GoEmotions: A Dataset of Fine-Grained Emotions. ACL. URL: https://arxiv.org/abs/2005.00547 (дата звернення 25.05.2025).

14. Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. URL: https://arxiv.org/abs/1606.05250 (дата звернення 03.06.2025).

15. Hugging Face T5 Documentation. URL: https://huggingface.co/docs/transformers/model_doc/t5 (дата звернення 02.06.2025).

16. Anaby-Tavor, A., et al. (2020). Do Not Have Enough Data? Deep Learning to the Rescue! URL: https://arxiv.org/abs/2003.02275 (дата звернення 03.06.2025).

17. Guo, M., et al. (2020). Nonlinear Mixup: Data Augmentation Beyond Linear Interpolation. URL: https://arxiv.org/abs/2010.02394 (дата звернення 04.06.2025).